a plurality of input/output devices coupled to the input/output interface, the plurality of processors processing program code configured to perform a plurality of tasks, the program code comprising:

program code configured to cause a first portion of the plurality of processors to interact with a first input/output device of the plurality of input/output devices;

program code configured to cause a second portion of the plurality of processors to interact with a second input/output device of the plurality of input/output devices;

program code configured to cause a second portion of the plurality of processors to emulate a specific microprocessor instruction set.

Claim 14 (currently amended): The apparatus of claim ~~8~~ 13 further comprising:

kernel program code configured to dynamically allocate the processing of the program code among the plurality of processors.

REMARKS

The Examiner's comments and the cited art have been noted and carefully studied. Applicant respectfully traverses the rejections and request reconsideration. For the reasons set forth below, Applicant submits the remaining claims are allowable as written.

§ 112 Rejections

Dependent Claim 6

Applicant submits an amendment to claim 6, without prejudice, and without narrowing over the unamended claim 6, to correct a typographical error by replacing the language "further wherein the tasks comprise" with "wherein the tasks comprise." Further, Applicants submit that the amended claim is fully supported by the specification and that the amended claim fully satisfies the requirements of 35 U.S.C. 112.

Dependent Claim 14

Applicant submits an amendment to claim 14, without prejudice, and without narrowing over the unamended claim 14, to correct a typographical error by replacing the language "The apparatus of claim 8" with "The apparatus of claim 13." Further, Applicants submit that the amended claim is fully supported by the specification and that the amended claim fully satisfies the requirements of 35 U.S.C. 112.

§ 102(a) Rejections

The Office Action rejected claims 1 under 35 U.S.C. 102(a) as being allegedly anticipated by Broder et al. (U.S. 5,991,808).

A claim is anticipated only if each and every element as set forth in the claim is found, either expressly, or inherently described, in a single reference.[1] Furthermore, the identical invention must be shown in as complete detail as contained in the claim.[2]

Applicants submit that Broder et al. fails to disclose each and every element of Applicants' claimed subject matter and respectfully request the Examiner to withdraw the rejections. In addition, Applicants submit that Broder et al. does not, disclose, teach or suggest, either implicitly or explicitly, Applicants' claimed subject matter.

Claim 1

Broder et al.

Broder et al. is directed to using load balancing to distribute workloads over networks of workstations or servers or mirrored sites, (col. 1, lns. 1-17). A client is used as a task directing unit that is interconnected to a plurality of resources, (col. 2, lns. 51-55), where the resources may be processing units that are all part of single multiprocessor computing device such as a high power workstation with multiple processors, or may be each part of a respective single processor computer, which together form a bank or network of servers or workstations, (col. 2, lns. 42-49). The task directing unit is configured to obtain load information of each of the resources selected uniformly at random, (col. 2, lns. 55-57). The task directing unit queries each of the resources for load information and each of the resources respond to the query with its loading information, (col. 2, lns. 58-63). The task directing unit identifies the least loaded of the resources selected for processing a task, (col. 3, lns. 28-37), by selecting them at random, and then determining which of such resources is least loaded. The task directing unit is programmed to operate in the manner described above, (col. 3, lns. 44-45).

---

[1] *Glaverzel Societe Anonyme v. Northlake Marketing & Supply, Inc.*, 75 F.3d 1550, 1554 (Fed. Cir. 1999); *Verdegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628, 631, 2 U.S.P.Q.2d 1051, 1953 (Fed. Cir. 1987).

[2] *Richardson v. Suzuki Motor Co.*, 868 F.2d 1226, 1236, 9 U.S.P.Q.2d 1913, 1920 (Fed. Cir. 1989).

Independent Claim 1

Sending A Task To Pre-Programmed Servers IS NOT identifying available processing resources ... [and] providing to the available processing resources functional programs and initial data corresponding to the tasks...

> Applicants submit that Broder et al. does not disclose, teach or suggest Applicants':
>
> "queuing tasks, identifying available processing resources in the homogeneous multiprocessor environment, allocating the available processing resources among the tasks, *providing to the available processing resources functional programs* and initial data corresponding to the tasks, and performing the tasks using the available processing resources to produce resulting data."

(claim 1). Applicants respectfully submit that Broder et al. simply discloses a load balancing system where identically loaded servers are each initially programmed to perform those tasks for which they may be assigned. Broder et al. makes reference to such operation throughout its disclosure where it simply discloses checking the load of its servers, and does not check what type of tasks that they are configured to perform. Further, once a server is identified as a server to receive a next requested task, Broder et al. discloses that it is at this point that the task is sent to the server, (e.g., col. 6, lns. 8-10), and nowhere is there discussed the sending functional programs along with the task. Thus, because Broder et al. never sends or provides a program to the available server to perform the given task, Broder et al. cannot, and does not, disclose, teach or suggest Applicants' claimed subject matter.

Applicants further note that Applicants' claim 1's claim language includes, inter alia, "identifying available processing resources," and "providing to the available processing resources functional programs." Such language indicates that the processing resources be first identified as being available before they can be provided with the functional programs. The fact that the servers in Broder et al. are capable of executing a given task, without having a corresponding program passed to such servers, indicates that such programs were known to be present before the identification of a server as being available, and as such not only does not disclose Applicants' claimed subject matter, but in fact, teaches away from such claimed subject matter.

Claim 13

Fitch et al.

Fitch et al. is directed to a system for process scheduling from within a current context and switching contexts only when the next scheduled context is different. A control context may

be identified as a thread, process or task, where a context is in control of a machine environment until it yields, is blocked waiting for something, or is interrupted (descheduling events), (col. 1, lns. 42-49). Fitch et al. discloses a parallel processing environment having multiple address spaces, (col. 3, lns. 53-55), each address space containing application code and kernel code, (Fig. 3), and the context scheduling management system described in Fitch et al. is directed to enhancing the processing within a single address space of the parallel data processing environment, (col. 3, lns. 63-66). Further, an address space can only execute one context (task) at any one time, and if a new context (task) arrives at the address space, the currently running context (task) determines whether the new pending context (task) has priority, and if it does, deschedules the currently running context (task) in favor of the new pending task, (col. 4. lns. 3-34). As such, the context scheduling management system is not a parallel processing scheduling system, even though it operates among many address spaces that are operating in parallel with one another.

Independent Claim 13

Applicants submit that Fitch et al. does not disclose, teach or suggest Applicants':

> "a plurality of processors coupled to a bus, *an input/output interface* coupled to the bus, a *plurality of input/output devices coupled to the input/output interface*, the plurality of processors processing program code configured to perform a plurality of tasks, the program code comprising: program code configured to cause a first portion of the plurality of processors *to interact with a first input/output device* of the plurality of input/output devices, program code configured to cause a second portion of the plurality of processors *to interact with a second input/output device* of the plurality of input/output devices; program code configured to cause a second portion of the plurality of processors to emulate a specific microprocessor instruction set."

(claim 13).

A Communication Interface Connecting Multiple Processing Nodes IS NOT a plurality of processors coupled to a bus, [and] an input/output interface coupled to the bus ...

Applicants submit that although Fitch et al. discloses a communication interface 16 connected between the processing nodes 12, (Fig. 1), that the communication interface 16 is limited to "bi-directional transfer of data," (col. 3, lns. 43-44), between the "four processing nodes 12," (col. 3, ln. 41). As such, rather than any input/output interfaces being coupled to a bus, Fitch et al. only discloses processors as being coupled to the communication interface 16. The adding of input/output interfaces to communication interface 16 would reduce its ability to

efficiently handle the data communications between the four processing nodes by at least increasing the traffic on such communications interface. Further, the absence of any input/output interfaces being disclosed in Fitch et al. further includes the absence of any input/output devices being disclosed as connected to such input/output interfaces. Applicants further submit that even though kernel 24 of a processing node 12 knows that it is only one of many other kernels 24 in a parallel processing environment 10, such parallel processing environment 10 is absent any input/output interfaces connected to input/output devices where such environment is shown as only having processing nodes 12, a CPU, communications, memory, and a clock. Thus, Fitch et al. is absent any disclosure of the coupling of both a plurality of processors and a plurality of input/output interfaces coupled to a bus, including the lack of any disclosure input/output devices being further connected to such input/output interfaces.

Segregating Process Operations On A Single Processor In Time IS NOT program code configured to cause a first portion of the plurality of processors to interact with a first input/output device of the plurality of input/output devices ...or... to cause a second portion of the plurality of processors to interact with a second input/output device of the plurality of input/output devices

Next Applicants directs the Examiner's attention to the statement within Fitch et al. that indicates that an operating system divides the machine environment managed by an operating system among different control contexts, where only one context or task can be running within any particular address space, having a single processor at any point in time, is an example of a dividing operations in time, rather than dividing operations between physical aspects of the parallel data processing environment, (col. 4, lns. 3-5). As such, Applicants submit that Fitch et al. is absent discussion disclosing or suggesting that particular processes are programmed to interact with any subset of any set of input/output devices, whether a first set of processors with a set of first input/output devices, or whether a second set of processors with a set of second input output devices.

§ 103(a) Rejections

Claims 2-3, 5-11, and 14

The Office Action rejects claims 2-3, 5-11, and 14 under 35 U.S.C. 103(a) as being allegedly unpatentable over Broder et al. in view of Fitch et al.

Dependent Claim 2

Applicants submit that although Broder et al., a reference disclosing a distributed environment, i.e., an environment where a single tasks can be routed to different processors based on availability, may make reference to the use of processors having different processors speeds, one of ordinary skill in the art would not modify Broder et al., with the knowledge of Fitch et al., to achieve a system where different processors execute different instruction sets.

Broder et al.'s discussion of using processors having different speeds include different processors loaded with the same programs, where the only difference between the processors is the speed at which they execute. Such a difference between processor operations is wholly different then where different processors are to handle different instruction sets as the logistics to handle different instruction sets at different processors is far greater than those to handle different processing speeds. Further, to the extent that Broader et al. only queries processes as to their current load rather than what instruction sets they handle, i.e., only shows concern for whether a process has available time for more work rather than what type of work the processor is capable of performing, Broader et al. teaches away from having different processors with multiple instruction sets.

In addition, Applicants further submit that Fitch et al., like Broder et al., does not disclose, teach or suggest the use of different instruction sets by different processors. Fitch et al. is directed to simply switching between different tasks rather than between different tasks having different instruction sets. Applicants submit that Fitch et al. is absent any discussion of loading a control context which also includes the capability of interpreting a different instruction set than that contained by a previously executing context.

Applicants submit that that the Office Action's arguments concerning obviousness are merely conclusory and do not show, at least, the requisite motivation to combine, whether from the references alone, the problem to be solved, or that generally known by one of ordinary skill in the art. Additionally, since none of the cited references teach or suggest a system wherein a plurality of processors of a homogenous multiprocessor environment are capable of executing a first instruction of a first instruction set and a second instruction of a second instruction set, the combination of any of the cited references cannot produce the Applicants' invention as claimed.

Further, Applicants submit that at least because claim 2 depends from claim 1, and as a dependent claim therefrom, claim 2 is allowable for the reasons claim 1 is allowable. Applicants

further submit that claim 2 is also allowable in light of the presence of novel and non-obvious elements contained in claim 2 that are not otherwise present in claim 1.

Dependent Claim 3

For similar reasons discussed above regarding claim 2, (i.e., that none of the cited references teach or suggest a system wherein a plurality of processors of a homogenous multiprocessor environment are capable of executing a first instruction of a first instruction set and a second instruction of a second instruction set), Applicants therefore further submit that neither reference alone, or in combination, disclose, teach or suggest a further first and second instruction from different instruction sets having identical bit patterns but which perform different operations.

Further, Applicants submit that at least because claim 3 depends from claim 2, and as a dependent claim therefrom, claim 3 is allowable for the reasons claim 2 is allowable. Applicants further submit that claim 3 is also allowable in light of the presence of novel and non-obvious elements contained in claim 4 that are not otherwise present in claim 2.

Dependent Claim 5

For similar reasons discussed above regarding claims 2 above, (i.e., that none of the cited references teach or suggest a system wherein a plurality of processors of a homogenous multiprocessor environment are capable of executing a first instruction of a first instruction set and a second instruction of a second instruction set), Applicants therefore further submit that neither reference alone, or in combination, disclose, teach or suggest a converting a functional program of the functional programs expressed using the first instruction set to an equivalent functional program expressed using the second instruction set.

Further, Applicants submit that at least because claim 5 depends from claim 3, and as a dependent claim therefrom, claim 5 is allowable for the reasons claim 3 is allowable. Applicants further submit that claim 5 is also allowable in light of the presence of novel and non-obvious elements contained in claim 5 that are not otherwise present in claim 3.

Dependent Claim 6

Applicants acknowledge the Examiner's comment that neither Broder et al. nor Fitch et al specify the systems for which they should be employed.

Applicants submit that at least because claim 6 depends from claim 3, and as a dependent claim therefrom, claim 6 is allowable for the reasons claim 3 is allowable. Applicants further

submit that claim 6 is also allowable in light of the presence of novel and non-obvious elements contained in claim 6 that are not otherwise present in claim 3.

Dependent Claim 7

Applicants submit that at least because claim 7 depends from claim 3, and as a dependent claim therefrom, claim 7 is allowable for the reasons claim 3 is allowable. Applicants further submit that claim 7 is also allowable in light of the presence of novel and non-obvious elements contained in claim 7 that are not otherwise present in claim 3.

Dependent Claim 8

Applicants submit that at least because claim 8 depends from claim 3, and as a dependent claim therefrom, claim 8 is allowable for the reasons claim 3 is allowable. Applicants further submit that claim 8 is also allowable in light of the presence of novel and non-obvious elements contained in claim 8 that are not otherwise present in claim 3.

Dependent Claim 9

Further, Applicants submit that at least because claim 9 depends from claim 8, and as a dependent claim therefrom, claim 9 is allowable for the reasons claim 8 is allowable. Applicants further submit that claim 9 is also allowable in light of the presence of novel and non-obvious elements contained in claim 9 that are not otherwise present in claim 8.

Dependent Claim 10

Further, Applicants submit that at least because claim 10 depends from claim 9, and as a dependent claim therefrom, claim 10 is allowable for the reasons claim 9 is allowable. Applicants further submit that claim 10 is also allowable in light of the presence of novel and non-obvious elements contained in claim 10 that are not otherwise present in claim 9.

Dependent Claim 11

Further, Applicants submit that at least because claim 11 depends from claim 8, and as a dependent claim therefrom, claim 11 is allowable for the reasons claim 8 is allowable. Applicants further submit that claim 11 is also allowable in light of the presence of novel and non-obvious elements contained in claim 11 that are not otherwise present in claim 8.

Dependent Claim 14

Applicants acknowledge the Examiner's statement that Fitch et al. does not specifically disclose the apparatus of claim 14.

Fitch et al. discloses the use of kernel code to control which task, of many queued tasks, has priority to run at any give time on a single processor. Further, it should be noted that Fitch et

al. discloses such use of kernel code to process tasks in a serial manner, even though Fitch et al. includes parallel processing. As such, Applicants submit that where Fitch et al. includes parallel processing, but limits its use of its kernel code to controlling non-parallel processing, and where such kernel code only is in control of a single processor, that Fitch et al. teaches away from Applicants' claimed subject matter.

Applicants further submit that neither Broder et al., which is absent any disclosure regarding the use of Kernel code, nor Fitch et al., which teaches away from Applicants' claimed subject matter, disclose, teach or suggest that such references could be combined to achieve the claimed subject matter of Applicants', and as such, Applicants' claim 14 subject matter is patentable in light or such references.

Further, Applicants submit that at least because claim 14 depends from claim 13, and as a dependent claim therefrom, claim 14 is allowable for the reasons claim 13 is allowable. Applicants further submit that claim 14 is also allowable in light of the presence of novel and non-obvious elements contained in claim 14 that are not otherwise present in claim 13.

Claim 4

The Office Action rejects claims 4 under 35 U.S.C. 103(a) as being allegedly unpatentable over Broder et al. in view of Fitch et al., in further view of Frankel.

Frankel et al.

Frankel et al. is directed to a real-time operating system and virtual digital signal processor for the control of a digital signal processor. The system utilizes abstract objects arranged in hierarchical fashion to enable a high level programming language to be used in accessing a wide variety of available functions. (Summary of The Invention). These hierarchical objects are supported by a real-time, multi-tasking system core that manages memory, interrupts, and task-switching. (Id.) The system also includes capabilities of performing I/O operations on an attached host processor file system. (Id). The user of the system is thus not required to be cognizant of the underlying hardware resources, but can program concisely in high level language to carry out operations in ways for which the DSP is best suited. (Id). Further, the system utilizes a kernel providing I/O communications and communication functions, but does not disclose executing such a kernel on multiple processors.

Dependent Claim 4

Applicant acknowledges Examiner's statement that neither Broder et al. nor Fitch et al. disclose the method of claim 4.

Although Frankel et al. discloses a kernel providing I/O communications and communication functions, it does not discuss a plurality of processors executing such kernels. Further, Fig. 1 indicates a lack of presence of multiple processors where only a single CPU 11 is shown. Further, as discussed above regarding claim 2, neither Broder et al., nor Fitch et al. disclose, teach or suggest the use of different instruction sets by different processors. Applicants submit that at least for the reason that none of the cited references teach or suggest a system wherein a plurality of processors executes a program or kernel including first and second instruction sets, the combination of any of the cited references cannot produce the Applicants' invention as claimed.

Applicant submits that at least because claim 4 depends from claim 3, and as a dependent claim therefrom, claim 4 is allowable for the reasons claim 3 is allowable. Applicant further submits that claim 4 is also allowable in light of the presence of novel and non-obvious elements contained in claim 4 that are not otherwise present in claim 3.

Claim 12

The Office Action rejects claims 12 under 35 U.S.C. 103(a) as being allegedly unpatentable over Broder et al. in view of Fitch et al., in further view of Hardwick.

Hardwick

Hardwick is directed to a dynamic load balancing among processors in a parallel computer. The system includes the determining, at a processor who has just received a task, the computational cost of executing such task, and if it is too expensive to ship it to another processor for execution. (Summary Of The Invention). The system is implemented using a nested parallel programming model called the team parallel model and supports a nested parallel language that can be used to implement irregular device-and-conquer programs on parallel computers.

Dependent Claim 12

Applicants submit that at least because claim 12 depends from claim 3, and as a dependent claim therefrom, claim 12 is allowable for the reasons claim 3 is allowable.

Applicants further submit that claim 12 is also allowable in light of the presence of novel and non-obvious elements contained in claim 12 that are not otherwise present in claim 3.

CONCLUSION

For the foregoing reasons, withdrawal of the rejections and allowance of the remaining claims is respectfully requested. If there are any questions or comments regarding this response, the Examiner is encouraged to contact the undersigned at 312-609-7500.

Respectfully submitted,

By: _____
Brent A. Boyd
Reg. No. 51,020

Dated: June 16, 2003

Vedder, Price, Kaufman & Kammholz
222 North LaSalle Street
Chicago, Illinois 60601
Telephone: (312) 609-7500
Facsimile: (312) 609-5005